

Joc per a dispositius Android amb l'ús de la geolocalització

Juan Ricardo Hernández Chaves

Resum—Aquest article descriu la creació d'un joc per a dispositius mòbils basats en Android. Geodyr és una aplicació que utilitza la geolocalització com a novetat davant la resta de jocs. Aquest és un joc que té per objectiu lluitar, capturar i col·leccionar els monstres que hi ha situats arreu del mapa real de Google Maps i que per a cada un d'ells s'haurà de fer una batalla, guanyar-la i poder arribar a ser el número 1 del rànking sempre i quan el jugador sigui el que més experiència tingui respecte la resta d'usuaris. Aquesta aplicació s'ha desenvolupat mitjançant eines com Eclipse fent ús del llenguatge Android per a dispositius amb una versió més gran o igual a la 4.2 (Jelly Bean). També, s'ha fet un estudi sobre la llibreria LibGDX, que es fa servir per desenvolupar jocs multiplataforma i que en aquest cas, ha servit per conèixer la seva utilitat una mica més a fons.

Paraules clau—Android, Geolocalització, Joc, Dispositiu, LibGDX, Eclipse, Gradle, Java.

Abstract—The aim of this article is to describe the making of a game designed for the android mobile operating system. Geodyr runs with the use of geolocation, an engine which works like a radar source, determining a meaningful location in a geographic area. This game also employs the Google Maps application; you can fight against the monsters located throughout the map, and collect them to upgrade your ranking. Geodyr has been developed by the use of Eclipse engine and it employs Android's official language for devices running Android 4.2 (Jelly Bean) or higher. It is noteworthy to mention the research that has been made of libGDX, an application framework written in Java for the development of desktop and mobile gaming systems.

Index Terms—Android, Geolocation, Game, Device, LibGDX, Eclipse, Gradle, Java.



1 INTRODUCCIÓ

Avui dia els telèfons mòbils són una tecnologia molt vigent a la nostra vida i estem habituats a fer-lo servir de manera quotidiana. Els factors que fan que sigui molt addicte no és només el fet de socialitzar amb la gent que ens envolta mitjançant Facebook, Twitter o WhatsApp, sinó que també el fet d'escollar música, veure vídeos o jugar són ingredients molt importants. Aquest últim és un punt clau sobre els dispositius, la cota de mercat és molt alta i els usuaris volen més jocs, de més qualitat i amb millors gràfics. Aquest treball tracta de la creació d'un joc que té com a objectiu capturar tots els monstres que apareixen en el mapa mitjançant una lluita prèvia fent ús dels monstres que s'han anat capturant. El projecte, que es basa en una versió preliminar amb una funcionalitat limitada, es vol aconseguir en aquest cas que en el joc es pugui lluitar, capturar i gestionar els monstres.

Aquest projecte és un joc per a dispositius Android el qual fa ús de la geolocalització per situar-nos pel mapa real de Google Maps. El joc consisteix en lluitar i

capturar monstres per cada punt localitzat en el mapa a la vegada que s'adquireix experiència i nivell tant pel jugador com pels monstres capturats amb l'objectiu de tindre'ls tots i arribar a ser el número 1 del rànking. Per aconseguir aquest monstres s'ha de lluitar amb els que tenim equipats d'aquells que s'han capturat abans depenent del criteri del jugador.

La principal dificultat d'aquest projecte és la programació de la batalla entre monstres amb la llibreria LibGDX[1]. La idea ha estat aprofundir en aquest llenguatge per a implementar jocs multiplataforma per dispositius mòbils i incrustar-lo dins de l'aplicació per tal d'interactuar amb la interfície nativa d'Android[2]. Finalment en aquest article s'explicaran tots els punts claus pel qual s'han anat passant fins a obtenir els resultats desitjats.

A continuació, s'explicaran els objectius proposats per a aquest projecte i els canvis que hi ha hagut, l'estat de l'art sobre els dispositius i eines de desenvolupament actuals, la metodologia emprada pel desenvolupament del joc, discussió dels resultats obtinguts i finalment les conclusions a què s'ha arribat.

-
- E-mail de contacte: juanricardo.hernandez@e-campus.uab.cat
 - Menció realitzada: Enginyeria del Software i Tecnologies de la Informació.
 - Treball tutoritzat per: Ramon Martí Escalé (Dept. d'Enginyeria de la Informació i de les Comunicacions).
 - Curs 2014/15

2 OBJECTIUS

A continuació, s'explicaran els objectius que aquest treball s'han portat a terme durant el llarg del projecte, com són: Els objectius de la infraestructura del joc (servidors, bases de dades, hosting), els objectius sobre l'aplicació (fins a on es vol arribar per poder jugar) i finalment l'objectiu pel qual es vol utilitzar LibGDX per desenvolupar la batalla entre monstres.

2.1 APLICACIÓ

L'objectiu principal és desenvolupar una aplicació que serà un joc que tractarà de trobar monstres situats a diferents punts del mapa (en aquest cas mapa de Catalunya), batallar amb ells, capturar-los i col·leccionar-los. Llavors, el que s'ha d'implementar en aquest projecte són les funcionalitats següents:

Objectius relacionats amb els monstres:

- Els monstres obtindran experiència conforme battleen amb altres i pujaran de nivell.
- Els monstres tindran habilitats pròpies i aquestes s'aniran aconseguint conforme pugin de nivell.
- Els monstres tindran la capacitat d'equipar objectes per millorar les seves habilitats.

Objectius relacionats amb el jugador:

- El jugador podrà tindre fins a un màxim de 4 monstres equipats.
- El jugador podrà seleccionar un monstre diferent al que ja té equipat per un altre.
- Per obtenir monstres, el jugador s'haurà de moure pel mapa mitjançant GPS o WiFi fins les coordenades indicades per cada un d'ells.
- El jugador podrà col·leccionar tots els monstres i podrà tindre més d'un monstre igual.

Al final del projecte el joc ha de poder caminar pel mapa real fins a les coordenades de cada monstre, poder battlear i capturar un monstre del mapa, poder pujar de nivell amb els monstres equipats i poder visualitzar la llista de rànkings d'experiència dels diferents usuaris.

Un altre de les idees principals és poder desenvolupar la batalla entre monstres amb LibGDX on l'objectiu és aprofundir en aquest llenguatge per programar jocs en dispositius mòbils i aprendre com funciona, poder crear exemples i entrar i conèixer les dificultats que ens proposa aquest món per a que després, un cop après les tècniques de desenvolupament poder-los aplicar a la batalla entre monstres i a més, capturar-los i fer-los servir més tard.

2.2 INFRAESTRUCTURA

Per poder fer funcionar els monstres pel mapa i tindre una base de dades funcional emmagatzemada, es va desenvolupar un servidor que tingués i suportés un servei Apache i MySQL (Phpmyadmin) per gestionar la base de dades. Gràcies a aquest servidor, el dispositiu es pot comunicar amb ell per obtenir les dades necessàries de la base de dades i mostrar-les per pantalla mitjançant arxius PHP, que donen suport a la comunicació entre el dispositiu

i la BD. Per assegurar la persistència de les dades del joc i dels usuaris, es va crear una base de dades amb taules per: Emmagatzemar les dades dels usuaris, les dades dels monstres i d'aquelles relacions que tenen entre elles, ja que per exemple, una habilitat pot ser assignada a més d'un monstre i es guarda una relació entre elles.

3 ESTAT DE L'ART

En aquest apartat, s'explicarà l'estat de l'art que actualment hi ha sobre Android i les aplicacions, la llibreria de LibGDX[3], les eines de desenvolupament que hi ha avui dia i de com podria ser els jocs en un futur no molt llunyà.

3.1 APLICACIONS ACTUALS

Actualment existeixen moltes aplicacions creades per smartphones amb sistema Android, com poden ser de jocs, d'utilitat, d'esports, de música, de llibres i moltes més. Però, aprofitant la tecnologia que incorporen els dispositius d'avui dia, es va pensar de crear un joc utilitzant la geolocalització, en que fins al moment, només existeix un de semblant a la idea proposada anomenat: Ingress. Aquest tipus de joc es basa en l'ús de la senyal de GPS (o geolocalització) aprofitant les eines que Google proporciona pel mapa de "Google Maps" i la realitat augmentada, descobrint llocs emblemàtics de cada ciutat mentre es lluita per la conquesta del terreny descobert. Donat que és complicat desenvolupar per a aquesta tecnologia, existeixen poques aplicacions dedicades a jocs d'aquest tipus i s'ha volgut innovar i crear una aplicació basada en el mateix sistema però amb una altra dinàmica de joc.

3.2 LIBGDX

En quant a la llibreria LibGDX, és un marc de joc multi-plataforma que suporta Windows, Mac OS X, Android, iOS i HTML5 i utilitza codi Java. Actualment es va actualitzar a una versió que fa és de Gradle[4] per compilar les aplicacions i que si es fa servir un projecte antic sobre eines de desenvolupament actuals amb Gradle, aquest no funcionarà i s'haurà d'adaptar.

3.3 EINES DE DESENVOLUPAMENT

Actualment, les eines que es poden fer servir per desenvolupar el joc, són:

- **Eclipse Android ADT 2013[5]:** Eclipse creat específicament per desenvolupar aplicacions Android (Actualment desfasat) amb tots els pluggins necessaris per poder programar i executar per a qualsevol versió de sistema.
- **Android Studio[6]:** Aquest any 2015 Google ha creat una eina estàndard de desenvolupament únicament per Android sense la necessitat d'instal·lar pluggins o altres addicions. En cas que l'eclipse no funcioni correctament, es tindrà com alternativa aquest software.

Són les eines més importants i preparades per programar en Android i que actualment es fa servir Android Studio.

3.4 FUTUR

La idea és que en un futur no molt llunyà es pugui crear aplicacions interessants amb la geolocalització i la realitat augmentada. Ja que avui dia encara sembla que costa que la realitat augmentada tingui protagonisme a les nostres vides, pot ser de molta utilitat combinar-lo amb la geolocalització i que per a jocs mòbils, pot donar molta potencia en el sector i que pot ser molt interessant si s'explota encara més, com per exemple: Caçar monstres en 3D, veure informació sobre objectes reals en 3D o que sigui d'utilitat en la aplicació. Simples idees que poden donar molt de joc.

4 METODOLOGIA

Per al desenvolupament d'aquest projecte, s'ha seguit una metodologia amb les següents tasques:

- **Planificació:** Planificació general del projecte on es defineix l'àmbit del projecte, estudi de viabilitat, anàlisi de riscos, cost del projecte, planificació temporal i recursos necessaris.
- **Anàlisi:** S'analitza els objectius principals del joc, cap a on es vol arribar, les eines que es faria servir pel desenvolupament i la documentació adequada per a les necessitats del projecte.
- **Disseny:** Definició de les pantalles que interactuen directament amb l'usuari com el disseny de les pantalles i el disseny de la batalla.
- **Implementació:** Fase de desenvolupament de la aplicació on s'implementa les funcionalitats de les pantalles i de la batalla.
- **Test:** Fase de realització de proves amb dispositius de diferent tipus i mides.

4.1 METODOLOGIA ÀGIL

Per aconseguir els objectius proposats, es va fer servir una metodologia àgil de desenvolupament com és *Scrum* (Mostrat a la figura 1):

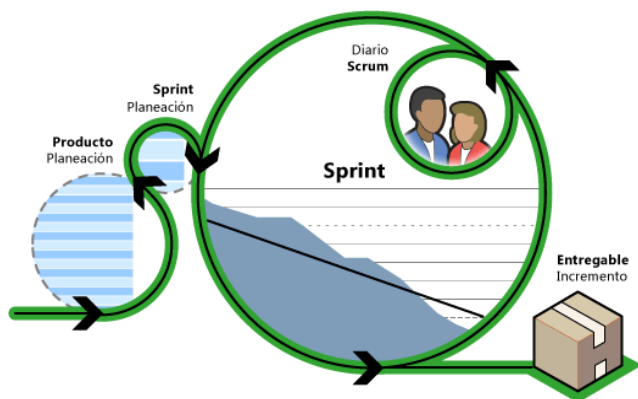


Figura 1: Diagrama mètode Scrum

A l'inici del projecte es va decidir que els dilluns de cada setmana es decidiria els objectius que s'havien de complir

pel pròxim dilluns i per tal de poder tindre un control de la feina feta, s'avaluàrien els objectius proposats i es definiria quins d'ells són més fàcils i quins són més difícils i quins han de prendre més atenció. A més a més, dues o tres setmanes abans de l'entrega, tot el que no s'hagi pogut complir, es descartarà i es centrarà en aquells que sí s'han complert i assegurar que el que s'ha fet tingui la qualitat desitjada mitjançant proves de l'aplicació. Finalment, la metodologia que s'ha anat seguint fins ara ha sigut la proposada al principi del projecte, com són:

- Realitzar una revisió setmanal per avaluar la feina realitzada.
- Realitzar un estudi previ abans d'implementar per no perdre temps durant la implementació.
- Assignar objectius cada setmana i portar una idea de la feina realitzada i la que falta per fer.

Tot i ser una sola persona el que realitza aquest projecte, s'ha volgut aplicar una metodologia semblant a Scrum, adequat per aquest tipus de treballs en el que implica el desenvolupament de software i una metodologia àgil.

No obstant, es va fer un petit canvi en la metodologia i és que l'avaluació de la feina realitzada es va avançar a divendres per comptes de dilluns. D'aquesta manera, els objectius proposats s'avaluen aquest dia per coincidir amb les entregues dels informes i així poder explicar en elles la feina feta fins al moment.

4.2 TASQUES

Per portar a terme aquesta metodologia, s'ha dividit les tasques en sis apartats: Planificació, Anàlisi, disseny, implementació i test. A continuació es detallarà cada una d'elles.

5 PLANIFICACIÓ

En la fase de planificació, el que es va fer abans de desenvolupar res, va ser aprendre com funciona la llibreria LibGDX: És un llenguatge dedicat pels jocs que s'ha de donar temps per tindre coneixements i aplicar-ho sobre aquest joc. Es varen fer exemples i llavors es va començar a desenvolupar l'aplicació d'aquells punts bàsics i generals, com són: La Interfície, les Pantalles i el Mapa. Fet els punts bàsics anteriors, es va començar a integrar LibGDX a la aplicació i a interactuar amb ella per poder batallar i capturar monstres.

A continuació es detalla els apartats el qual formen la fase de planificació del projecte:

- **Àmbit del projecte:** Es determina l'àmbit en el qual es vol enfocar el projecte.
- **Estudi de viabilitat:** Es fa un estudi sobre si el projecte que es vol desenvolupar és viable segons el temps, recursos i costos que suposa crear el joc.
- **Anàlisi de riscos:** En cas de no funcionar alguna de les parts importants, saber com reaccionar i per quin camí s'ha de continuar.

- **Cost del projecte:** Determinar si el projecte suposa un cost addicional o no.
- **Planificació temporal:** Realitzar un diagrama de Gantt i especificar les tasques i temps que es necessita en cada una d'elles al llarg del projecte.
- **Recursos necessaris:** Estudi dels recursos que es necessita pel poder realitzar el projecte.

Per imprevistos comentats en els resultats del apartat 10.1 i 10.2 amb LibGDX es va haver de que canviar la planificació de l'apartat d'implementació, tal i com es veu a la taula 1, sense interferir en la resta de tasques definides dins de les dates previstes:

DATA	TASCA
02-03-2015 fins 20-03-2015	Estudi sobre LibGDX: Com funciona, com es programa i com s'executa.
23-03-2015 fins 03-04-2015	Implementar la interfície.
06-04-2015 fins 01-05-2015	Implementar les funcionalitats de cada apartat de la interfície.
04-05-2015 fins 22-05-2015	Implementar la batalla entre monstres.

Taula 1: Planificació inicial

Un cop assabentat de que no es podria continuar el projecte amb l'objectiu de fer la batalla amb LibGDX es va decidir de canviar el sistema per desenvolupar-la amb codi natiu i tindre la batalla perfectament integrada com la resta de les parts de la interfície de la aplicació.

A continuació es mostra a la taula 2 els canvis que es va fer en la planificació per tal de continuar amb el desenvolupament sense que la resta de tasques quedés interferida i mantindre així el rang de dates previstes:

DATA	TASCA
02-03-2015 fins 20-03-2015	Estudi sobre LibGDX: Com funciona, com es programa i com s'executa.
20-03-2015 fins 03-04-2015	Implementar interfície i aprenentatge del LibGDX.
06-04-2015 fins 17-04-2015	Incorporar i executar els projectes entre l'aplicació i el LibGDX.
20-04-2015 fins 01-05-2015	Redisseny de la batalla de monstres i implementació de les funcionalitats.
04-04-2015 fins 22-05-2015	Implementació de la batalla.

Taula 2: Planificació final

Aquest canvi va ser l'únic que es va fer en tota la planificació del projecte ja que la resta, es va anar seguint segons les dates previstes fins arribar a test.

6 ANÀLISI

Abans de començar a desenvolupar el joc, es va fer una anàlisi del projecte i es va fixar: els objectius principals del joc, cap a on es vol arribar, les eines que es faria servir pel

desenvolupament, la documentació adequada per a les necessitats del projecte i les fonts d'informació respecte al contingut.

La idea general del projecte era continuar amb el que ja es tenia fet prèviament, ja que aquest projecte, com s'ha dit en altres casos, és tant ambiciós que no es pot fer tot a la vegada, de fer pas per pas i que en aquest cas, el projecte actual constaria de realitzar els canvis a la interfície d'usuari, pantalles informatives relacionades amb els monstres i enfocar l'atenció directament a la batalla.

En aquest apartat va donar lloc una anàlisi complet del projecte, com són:

- **Requeriments del projecte:** Es va analitzar quins eren els requeriments que es necessita mínimament per tindre un joc jugable. Ja que aquest projecte ja ve d'una versió prèvia, el que es va fer va ser agafar els requeriments dels quals ja es tenia fixats i es va seleccionar quins s'havien d'aprofundir en aquest treball: Fixar-se en els canvis de la interfície d'usuari i adentrar-se a la creació de la batalla entre monstres, ja sigui mitjançant LibGDX o codi natiu d'Android.
- **Metodologia emprada:** Saber quina metodologia s'utilitzarà al llarg del projecte per portar a terme cada una de les tasques sense problemes de continuïtat.
- **Objectius mínims o bàsics:** S'analitzen quins són els objectius mínims que s'ha de portar a terme per a que el joc tingui una mínima funcionalitat necessària i que es pugui jugar.
- **Abast del joc:** Analitzar fins a on es vol arribar amb aquest joc, per un costat per presentar davant el tribunal i per l'altre, fins a on es vol arribar després de la presentació del mateix. És una aplicació complexa i ambiciosa que es necessita molt de temps per portar-lo a terme i acabar-la.

Respecte a la batalla, es va fer una anàlisi de les possibilitats de crear-la amb un llenguatge fet expressament per programar videojocs per dispositius mòbils. Es va arribar a la conclusió de que LibGDX, un llenguatge preparat per a fer jocs per a dispositius mòbils, era la idea més viable i la qual es podia fer servir per integrar-la. En cas de que no fos possible desenvolupar amb aquest, la segona opció de la creació de la batalla més viable i directe era desenvolupar mitjançant el codi natiu Android, que precisament és com es va crear la resta de l'aplicació.

7 DISSENY

En la fase de disseny es va fer el menú de la interfície del joc, el qual serveix per a que l'usuari pugui navegar per les pantalles del joc, disseny de les funcionalitats analitzades prèviament en els requeriments, és a dir, les pantalles pel qual l'usuari interactuarà per poder jugar, disseny de la base de dades on es guarda la informació dels usuaris i els monstres i un servidor adequat que suporti el

tràfic de dades. Per dissenyar la estètica de la interfície i il·lustracions originals dels monstres, s'ha fet servir:

- **ADOBE Photoshop CS4** per definir les imatges de la interfície de l'aplicació.
- **Adobe Photoshop CS6** per il·lustrar les imatges dels monstres creades i el logotip fetes expressament per aquest joc.

Per aportar originalitat al joc, les il·lustracions de cada monstre han sigut realitzades per una persona especialitzada en el tema, estan fetes a mà amb photoshop i no existeix cap d'igual enlloc. A la figura 2 es mostra un exemple de monstre fet a mà que és com un ocell àguila:



Figura 2: Disseny d'un monstre

També hi ha d'altres il·lustracions que es van fer per tindre un marge més ampli de monstres cara a la lluita i cara al mapa, d'aquesta manera hi ha més varietat i sobretot originalitat (Es poden veure la resta d'il·lustracions dissenyats a l'apèndix A5). Els monstres que han sigut creats per aquest joc són els següents:

- **Drag blau:** Un monstre d'una gran mida, de color blau i que imposa davant qualsevol monstre.
- **Àguila imperial (Figura 2):** Un monstre volador que no es deixa intimidar per ningú.
- **Lleó blanc Tyr (Apèndix A5.1):** Lleó mitològic amb ganes de guanyar a qualsevol monstre del mapa.
- **Fada Myrna (Apèndix A5.2):** Monstre èpic senzill que s'obté només començar el joc per habitar-se a ell.

Per altre costat, s'ha reutilitzat imatges d'altres monstres que s'han buscat per Internet per així donar un volum més ampli al joc i tindre una vista més general del que es vol aconseguir a l'hora de voler col·leccionar tots els monstres del joc. A continuació es detallen les tasques que es va fer sobre la interfície, la infraestructura, la base de dades i el joc:

- **Disseny del logotip:** Es va tornar a dissenyar el logotip del joc per donar encara més sensació d'un joc per a dispositius mòbils. Abans es va fer un logotip encara més seriós i poc adequat a un joc on la prioritat és caçar i col·leccionar monstres situats pel mapa, així doncs es va crear un de més vistós i més amigable.
- **Disseny de la interfície:** Es va dissenyar cada pantalla del que està compost el joc detallant cada funcionalitat i interaccions amb l'usuari (per exemple el menú que es veu a la figura 3), com són les pantalles de: Mapa de distribució de monstres (com es veu a la figura 4), Equipar i desequipar Monstres que hi han capturades, informació completa individual de cada monstre, informació general de cada monstre en una llista col·leccionable i la batalla que hi haurà per poder matar, capturar i pujar de nivell per continuar endavant i batallar amb monstres encara més forts.

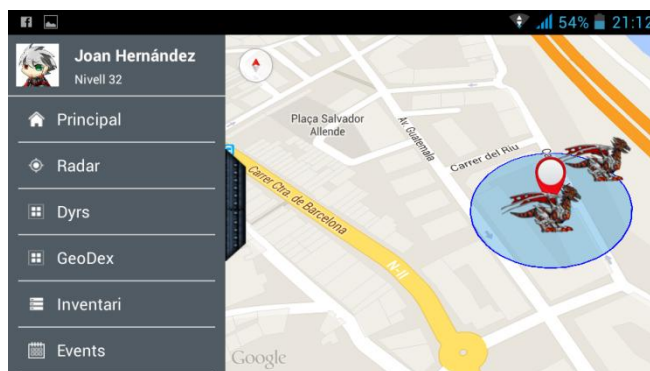


Figura 3: Navegació per l'aplicació

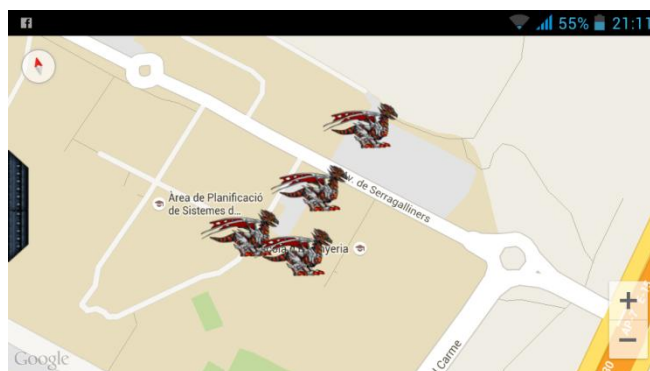


Figura 4: Mapa de monstres localitzats

- **Disseny de la infraestructura:** Abans d'emmagatzemar dades i arxius, es va dissenyar com havia de ser el servidor per poder donar els serveis desitjats. Finalment, es va obtenir un servidor hosting a Webempresa[7] amb un servei bàsic d'arxius per allotjar els serveis PHP que serveix

per la comunicació entre la base de dades i l'aplicació.

- **Disseny de la base de dades:** Per poder donar un servei de persistència al joc, es va dissenyar les taules de la base de dades que contindria les dades de cada usuari, de cada monstre, els monstres equipats o desequipats i de les posicions del mapa.
- **Estructura BD:** A més de les taules pròpies, s'havia d'estructurar com i quines taules afectarien a l'aplicació i si s'ha de crear altres com a suport i mostrar les dades de forma eficient (òptima) per després descarregar-les al mòbil sense sobrecàrrega de tràfic de dades. Aquestes taules són: monstres, usuari, usuari - monstre, habilitats i monstres - habilitats.

Finalment, respecte al disseny de les pantalles, es va tornar a realitzar a fer prototipus de les que es volien modificar per un disseny encara més amigable pel joc, com es pot observar a la figura 5. Es va anar dibuixant en paper diferents prototipus fins arribar al disseny definitiu. A continuació es mostra un exemple del que es va fer:

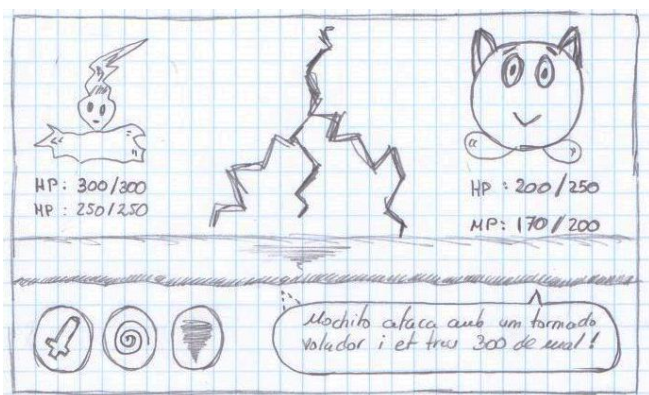


Figura 5: Prototip de la batalla

Per a més prototipus, a l'última part del document es pot veure un altre captura (Apèndix A4) sobre els monstres equipats.

8 IMPLEMENTACIÓ

En aquesta fase es va fer la implementació del joc, tant de les funcionalitats de cada pantalla com de la interfície d'usuari per navegar entre les pantalles. Per desenvolupar la batalla finalment s'ha fet servir també Eclipse ja que la versió preliminar del projecte es va fer amb aquesta eina i que, donat que amb Android Studio fa servir Gradle per compilar els seus projectes i Eclipse no, es va prendre la decisió de fer servir Eclipse i continuar fins al final. Per arribar a aquesta decisió, es va estar, segons la planificació, mitja fase investigant, fent proves i aprenent la funci-

onalitat de LibGDX i l'altre mig (havent modificat posteriorment la planificació) es va dedicar a implementar la batalla mitjançant Android.

8.1 CONFIGURACIÓ

Pel que fa a la configuració de la implementació, abans de realitzar res es va assignar noms comuns per a cada objecte de l'aplicació (Botons, imatges, taules, etc) com per exemple: Btn_habilitat1 o Btn_item1. Tan mateix, es va dividir per blocs cada funcionalitat en paquets diferenciats a Eclipse per portar a terme una organització lògica:

- **Paquet de Geolocalització:** En aquest paquet s'impliquen totes aquelles classes relacionades amb el mapa i el GPS. En aquest cas és la pantalla el qual es pot navegar pel mapa i trobar tots els monstres situats en cada punt establert a la BD.
- **Paquet de Base de dades:** En aquest paquet s'impliquen totes aquelles classes que donen suport a la base de dades. Es divideix en dos fases: En primer lloc cada cop que es fa login, aquest es comunica amb la base de dades i es descarrega en local tota la informació que no existeixi. En segon lloc, les pantalles que precisen d'un canvi a la BD al instant, hi ha una connexió directa per modificar les dades.
- **Paquet d'interfície d'usuari:** En aquest paquet s'impliquen totes aquelles classes que tenen a veure amb el menú o la navegació entre les diferents pantalles i el contingut de cada una d'elles.
- **Paquet comú:** En aquest paquet s'impliquen totes aquelles classes que tenen a veure amb pantalles o funcionalitats comunes, com per exemple la pantalla de login o de càrrega.

Referent a les imatges de l'aplicació, es va aplicar diferents resolucions per dal que de per cada dispositiu, tingués les mides ideals segons la pantalla i no es vegi amb una mida fixa ja que aquest joc s'està desenvolupant per a un dispositiu mòbil amb una pantalla de 5" i amb una qualitat en format PNG. En aquest format, per molta resolució que tingui el dispositiu, la imatge no es veurà afectada al ser augmentada.

En quant a la codificació, per donar un estil de codi apropiat al desenvolupament, es va fer servir Checkstyle[8], una eina de codi obert per ajudar a assegurar-se que el codi Java s'adhereixi a un conjunt d'estandards de codificació i el que fa és una inspecció del codi font i assenyalat els elements que s'aparten d'un conjunt definit de regles.

8.2 DESENVOLUPAMENT

En aquest apartat es va fer el desenvolupament general del joc, com són les funcionalitats relacionades amb la interfície i de la batalla:

- **Estudi del LibGDX:** Estudi exhaustiu del llenguatge LibGDX, que és on la majoria del temps es va portar per aprendre i poder desenvolupar la batalla entre monstres. Es van fer diferents exemples amb LibGDX amb ajuda de Makigas[9] (que es poden veure als annexes) amb diferents funcions bàsiques els quals es podien modificar per adaptar-se a la batalla de monstres, com: Moure o desplaçar objectes, afegir text a qualsevol part de la pantalla o crear un menú animat.
- **Implementar la interfície:** Es va desenvolupar la interfície del joc pel qual l'usuari pot navegar entre les pantalles importants per poder jugar. En aquest cas, s'ha refet les pantalles de mostra dels monstres equipats, la pantalla informativa d'un monstre en concret i el rànking de jugadors.
- **Implementació de les funcionalitats:** Un cop fetes les interfícies, es va implementar les funcionalitats que calen pel correcte funcionament del joc per cada pantalla com mostrar els monstres equipats de diferent estil o poder eliminar un monstre capturat.
- **Implementació i integració de la batalla:** A base de l'estudi fet prèviament, es va implementar la batalla amb totes les funcionalitats esmentats i crear les imatges dels monstres: Batallar contra el monstre del mapa, si el monstre enemic mata al nostre, que aparegui el següent equipat, poder atacar amb les 3 habilitats possibles amb un límit d'usos i finalment poder guanyar i capturar el monstre. En el cas de que l'enemic mati a tots els monstres equipats, la batalla haurà acabat perdent tot el progrés.

8.3 COMUNICACIÓ AMB EL SERVIDOR

Pel que fa a la comunicació amb la base de dades, el que es fa servir és JSON[10]. Directament, com es pot veure a la figura 6, tenim al servidor uns arxius PHP el que mitjançant un script es fa la consulta a la base de dades i retorna els valors mitjançant enllaços de codi JSON i aquestes dades són cridades pel codi Java dins de l'aplicació. Un cop que es rep les dades, es separa el contingut i se li assignen en variables lògiques que després serveixen per mostrar la informació demanada.



Figura 6: Enviar i rebre peticions mòbil - servidor

En el següent exemple es pot veure la crida a les dades des del joc i el que fa és demanar les dades cridant els arxius del servidor mitjançant una URL interna:

```
/** Constant que conté la url del script que utilitza el servei. */
public static final String URL_DADES_DYR =
"http://www.sandanime.com/webservice/dadesDyr.php"
;
```

```
JSONObject json = jsonParser.makeHttpRequest(
    BdUtils.URL_DADES_DYR, "POST", params);
```

```
dyrsUsuari =
json.getJSONArray(BdUtils.TAG_DADES_DYR);
```

En el següent exemple el que es fa és definir la URL on es te el fitxer amb les consultes de SQL i després mitjançant l'objecte de JSON a Java, el crida i envia la petició al servidor. Mentre que des del servidor, un exemple de que envia cap a l'aplicació és:

```
$query = " SELECT * FROM dyr";
try {
    $stmt = $db->prepare($query);
    $stmt->execute();
} catch (PDOException $ex) {

    $response["message"] = "Database Error1.
Please Try Again!";
    die(json_encode($response));
}
$response["message"] = "JSON ok";
$response["success"] = 1;
$response["dyrs"] = $stmt->fetchAll();
die(json_encode($response));
```

En aquest cas, el que fa l'script del PHP del servidor és preparar la consulta a la base de dades, l'executa i segons si ho ha fet be o malament, envia un missatge de confirmació i darrerament un missatge amb les dades obtinguts de la base de dades parsejades segons el nom de cada variable definides anteriorment.

És un mètode que realment surt eficaç si les dades que s'envien entre aplicació i servidor no són gaire grans i es manté una comunicació estable sense pèrdua de dades inclús si en comptes de fer servir Wi-Fi es fa servir 3G. En aquest cas només es realitzen consultes a la base de dades com per exemple veure els monstres equipats del jugador, afegir registre de monstre capturat un cop

batallat, veure la informació d'un monstre i mostrar-la per pantalla o eliminar un monstre que s'hagués capturat.

8.4 PROVES DURANT EL DESENVOLUPAMENT

Finalment, per provar que cada mòdul funcionés adequadament, es va provar l'aplicació en un telèfon mòbil Bq Aquaris 5 amb el sistema operatiu Android versió 4.2.2. La manera ideal d'haver provat la funcionalitat del joc tenia que haver sigut amb un emulador que simulés un telèfon mòbil, però el PC amb que es va desenvolupar el projecte, no tenia la capacitat necessària per executar-lo. Donat que no era possible, es va decidir fer servir un telèfon mòbil real, connectat per USB en mode desenvolupament on Eclipse ja porta incorporat el pluguin necessari per detectar el dispositiu.

9 TEST

En aquesta fase es va fer la realització d'un test exhaustiu de l'aplicació i confirmació que tot el que s'hagi implementat funcioni correctament.

Es va fer les proves adients al joc executant-lo en altres dispositius i així poder veure que tot funciona com s'espera:

- **Proves amb usuaris reals:** Un cop implementat el joc, una sèrie de persones es varen dedicar a provar l'aplicació i van donar el seu suport explicant errors i possibles millores.
- **Proves posicionament GPS:** Es va provar que el GPS tingui bon posicionament i precisió respecte al mapa del joc en tots els dispositius els quals van provar els usuaris.
- **Proves amb dispositius reals:** Ja que no es disposa d'un ordinador suficientment potent per poder testear el progrés de l'aplicació en un emulador, es va fer servir el telèfon mòbil particular del alumne.
- **Proves amb emuladors:** No obstant, per poder comprovar altres resolucions de diferents dispositius, es farà servir emuladors de diferents mides tot i no ser massa eficient.

Mentre que els usuaris que van testear anaven enviant els errors i idees que podien donar un major efecte positiu al projecte, es van anar corregint els errors en el codi i afegint les noves idees creant noves versions els quals els usuaris havien de tornar a instal·lar però sense tindre que esborrar res, ja que existeix un arxiu que guarda les dades locals de l'usuari com el login (usuari i contrasenya), d'aquesta manera s'evita un treball major i que a més, és més còmode pel que el fa servir.

9.1 PROBLEMES DETECTATS I ARREGLATS

Durant les proves, es van detectar una sèrie de problemes que es van tindre que anant arreglant pel correcte funcionament del joc:

- De vegades just després de fer login, el joc no carregava les dades correctament.
- A l'hora de realitzar la batalla, segons la resolució de la pantalla del dispositiu, l'aplicació es tanca-va a causa de no poder carregar les imatges.

A part dels errors de programació, es van donar idees respecte a algunes imatges que no quedaven gens be, com per exemple el fons de pantalla sobre la batalla o el login i fins hi tot crear un logotip nou tal i com es pot veure a l'apèndix A3.

10 DISCUSSIÓ DELS RESULTATS

Durant el projecte, la integració dels projectes fets amb LibGDX no va funcionar com s'esperava perquè a l'hora de compilar, Eclipse no podia incloure aquest projecte. Després de molta investigació per trobar una solució a aquest error, es va saber que desenvolupar amb LibGDX significava crear una aplicació el qual fa servir Gradle per compilar, adequat a les versions actuals del SDK d'Android (API 22 o més) mentre que l'aplicació preliminar no fa servir aquest sistema i que fa servir una versió inferior d'Android a la que es demana (API 19 o menys).

A continuació, s'explicaran els resultats que s'han obtingut al final d'aquest projecte, així com les eines de desenvolupament utilitzats, els resultats finals de la batalla i un canvi de servidor inesperat.

10.1 ECLIPSE, LIBGDX I CANVI DE VERSIÓ

Al llarg del projecte, es va veure que l'Eina de desenvolupament ideal va ser continuar programant amb Eclipse, perquè el projecte preliminar va ser creat fa un any enrere i a diferència amb l'actualitat, van canviar mètodes com la manera de compilar un projecte. Actualment, un projecte fet per Android Studio es compila mitjançant Gradle i en canvi un altre fet per Eclipse no. A conseqüència, no es va poder migrar l'actual projecte d'Eclipse a Studio. A més, a causa d'aquest canvi de compilacions, LibGDX també fa servir Gradle per compilar i voler ajuntar els dos projectes resulta incompatible.

A partir d'ara, els projectes que es desenvolupen amb Gradle i a diferència d'abans, aquest sistema aporta molts beneficis:

- Facilita la reutilització de codi i de recursos.
- Facilita la configuració, la extensió i la personalització de processos.
- Facilita la distribució de codi i llavors treballar en equip.
- Gestiona les dependències de forma més còmode i potent.
- Permet compilar des la línia de comandes i corregir errors.

- Facilita molt crear diferents versions del projecte.

10.2 RESULTAT DE LA BATALLA

D'un principi, en els objectius es va comentar que es faria servir Eclipse com a eina principal de desenvolupament (tant per l'aplicació com per la batalla entre monstres), però va haver un inconvenient a l'hora de fer-lo servir incloent LibGDX i es va haver de canviar de mètode. En conseqüència es va dividir la feina en dues parts:

- Per realitzar la interfície d'usuari de l'aplicació es feia servir Eclipse.
- Mentre que per realitzar la batalla entre monstres es faria servir Android Studio.

El problema va vindre a que fer servir LibGDX es necessita una versió SDK d'Android superior a l'actual i això interfereix el codi fet amb Eclipse, perquè al aplicar les actualitzacions necessàries es corromp i impedeix continuar. Degut a aquest problema, es va plantejar el desenvolupament amb dues eines independents.

La idea va ser que un cop s'hagués programat la batalla amb Android Studio, s'intentaria incorporar el codi dins de l'Eclipse. Però després de molta investigació i hores dedicades, es va poder adaptar LibGDX dins de l'Eclipse modificant la versió de l'SDK d'Android Studio a una versió més antiga. És a dir, el projecte de l'aplicació i del joc (la batalla) amb LibGDX són ara a la mateixa eina de desenvolupament (Eclipse ADT). No obstant, continua existint un problema greu i és: l'adaptació dels dos projectes units entre si. A l'hora de cridar el joc des de l'aplicació, no pot trobar els fitxers que toquen, tot i que s'indiqui prèviament mitjançant codi i configuracions.

Al donar moltes voltes al tema, es va provar de solucionar el problema amb altres mètodes i amb una es va saber que la versió més actual d'Android utilitza un sistema de compilació diferent, és a dir, LibGDX i Android API 22 o més utilitzen Gradle per compilar, mentre que el projecte Android actual (amb API 19) no compila de la mateixa manera. Al haver la diferencia de versions, el projecte amb LibGDX no compila i per tant, no funciona.

Finalment, com a nou objectiu en la fase de desenvolupament, es va descartar implementar la batalla amb LibGDX i per tant l'aplicació es va desenvolupar finalment amb codificació Java Android natiu entre les versions API 17 i API 19 (per dispositius amb Android 4.2 i Android 4.4) amb Eclipse com a eina principal, dedicant tot l'esforç en aquest punt. Finalment, tal i com es mostra a la figura 7, com a resultat final la batalla va quedar de la següent manera amb el codi natiu d'Android com a desenvolupament oficial:

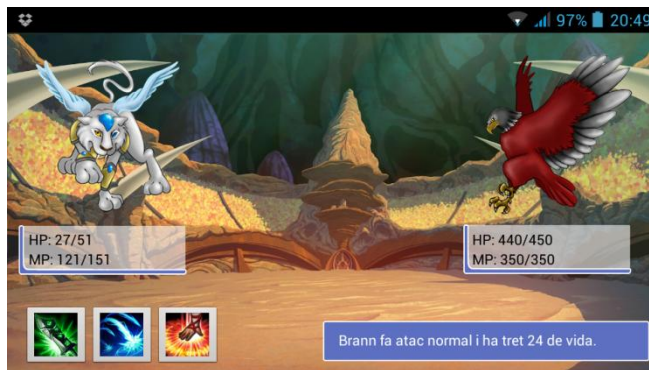


Figura 7: Batalla entre monstres

10.3 CANVI DE SERVIDOR

Cap al final del desenvolupament, el servidor el qual hi havia totes les dades va deixar de funcionar i per tant, es va haver de canviar i tornar a refer la base de dades i el hosting d'arxius. Aquest canvi, no va afectar pas a l'aplicació, doncs aquest té un sistema que mitjançant arxius PHP i comunicació via JSON, qualsevol servidor pot donar servei al joc en qualsevol moment i totalment transparent a l'usuari.

11 CONCLUSIÓ

Com a conclusió final es pot dir que s'han aconseguit realitzar els objectius que es van marcar al principi del projecte, que era continuar el desenvolupament del joc creant la batalla entre monstres, canviar la interfície d'usuari per un de més intuïtiu, afegir més funcionalitats i finalment donar més originalitat al joc: Canvi d'il·lustracions dels monstres o un logotip encara més amigable.

Gràcies a aquest treball, s'ha pogut continuar desenvolupant el joc tan com es volia ja que en aquest projecte el que es pretenia era ampliar coneixements sobre el món del desenvolupament de videojocs per a Android mitjançant la llibreria LibGDX i ampliar coneixements. Saber com funciona, com s'implementa i com es compila formen part dels objectius. Tot i els petits contratemps mencionats en els apartats anteriors sobre l'objectiu inicial de crear una batalla entre monstres, aquesta ha sigut realitzada i el joc assoleix la mínima funcionalitat que es demanava al principi del treball. Tot i així, s'ha pogut crear exemples funcionals de LibGDX amb diferents proves i algun "mini joc" (Veure apèndix A1). A més, s'ha pogut saber què és Gradle i el per què del que no es pugui compatibilitzar entre els projectes antics i nous. Tot i així, aquest és un projecte molt ambiciós i que, després de finalitzar el treball Fi de Grau, es farà el traspàs a una nova versió adequada a la tecnologia actual mitjançant Android Studio i es continuarà desenvolupant fins arribar a una versió definitiva i donar-se a conèixer al públic.

11.1 AMPLIACIONS FUTURES

De cara al futur, el que es pretén amb aquest projecte és continuar desenvolupant fins arribar a tots els objectius proposats inicialment amb dos companys més que, el que es vol és arribar a fer-ho públic i anar millorant conforme es va perfilant i adaptant a versions actuals i crear una petita empresa dedicat a aquest i futurs jocs o aplicacions especialitzats en geolocalització i realitat augmentada. És un projecte molt ambiciós, que consta de moltes funcionalitats i dissenys i que atraure al públic serà difícil però no impossible.

AGRAÏMENTS

Vull donar els agraïments a les persones que m'han donat idees i inspiracions per continuar aquest projecte tan gran i que tant m'agrada, com són: Sandra, per fer il·lustracions dels monstres que hi ha pel mapa i el qual es pot observar quan es captura a més del logotip del joc; Cristian Tanas per l'ajuda sobre els dubtes amb Android i la manera de compilar; Ramón Martí, per l'atenció i l'ajuda sobre el projecte; I a tots els amics que han testejat i opinat sobre el joc.

BIBLIOGRAFIA

- [1] LibGDX, Repositori de software oficial <https://github.com/libgdx/libgdx> Vist per última vegada el 16 d'Abril del 2015.
- [2] Google Develop, Android for developers <http://developer.android.com> Vist per última vegada el 19 d'Abril del 2015.
- [3] BadLogicGames, Android Java game development framework per LibGDX <http://libgdx.badlogicgames.com> Vist per última vegada el 16 d'Abril del 2015.
- [4] Gradle, Open Source Build Automation <https://gradle.org/> Vist per última vegada el 27 d'Abril del 2015.
- [5] Eclipse ADT, Repositori d'Eclipse Android desfasats <http://downloads.puresoftware.org/files/android/adt-bundle> Vist per última vegada el 10 de Juny del 2015.
- [6] Android Studio, Eina de desenvolupament Android <https://developer.android.com/sdk/index.html> Vist per última vegada el 11 de Juny del 2015.
- [7] Hosting server, Web empresa <https://clientes.webempresa.com> Vist per última vegada el 24 de Juny del 2015.
- [8] Check Style, Estil de codificació estàndard <http://eclipse-cs.sourceforge.net/> Vist per última vegada el 20 de Juny del 2015.
- [9] Makigas, Videotutorials sobre aplicacions Android Makigas <http://www.makigas.es> Vist per última vegada el 16 d'Abril del 2015.
- [10] JSON, Javascript Object Notation <http://json.org/> Vist per última vegada el 12 de Maig del 2015

APÈNDIX

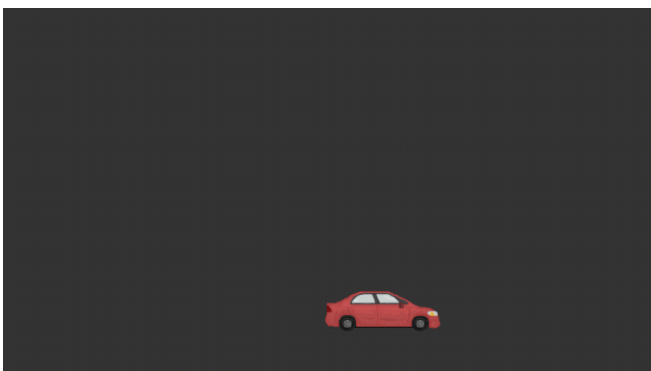
A1. EXEMPLES D'APLICACIONS EN LIBGDX

A continuació es mostren diferents exemples sobre aplicacions fetes:



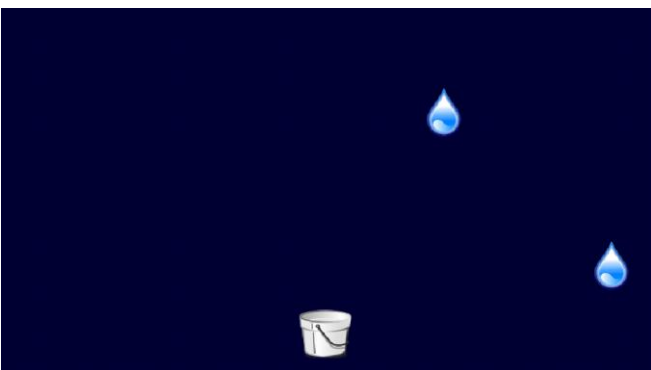
A1.1: Menú per seleccionar joc

Aquest primer cas és un clar exemple de com es pot crear un menú mitjançant LibGDX per tal de desplaçar-se entre diferents pantalles o aplicacions.



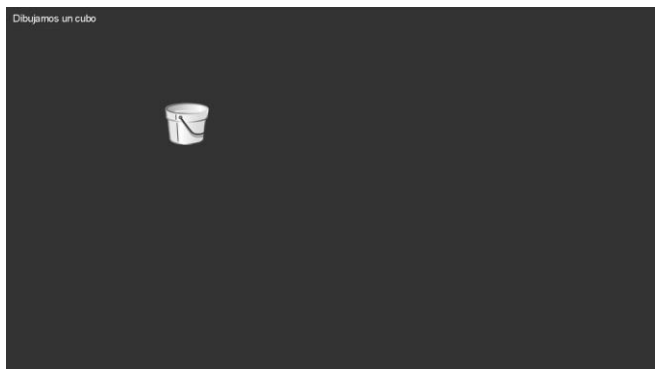
A1.2: Arrancar cotxe amb tàtil

Aquest segon cas és un exemple el qual fa ús de la pantalla tàtil per moure el cotxe cap a la dreta o cap a l'esquerra a una velocitat de menor a major segons es mantingui clicat.



A1.3: Recollir l'aigua amb el cub (tàtil)

En aquest tercer exemple, el mini joc tracta de tindre que recollir les gotes d'aigua i aquella gota que s'escapi, es finalitza la partida.



A1.4: Textos i moure el cub amunt i avall

En aquest quart exemple, el que es tracta és d'escriure paràgraf en les coordenades de la pantalla que sigui necessari. A la vegada, el cub es pot moure lliurement per la pantalla sense que hagi cap tipus de conflicte entre diferents objectes.

A3. LOGOTIP DE L'APLICACIÓ

Canvi de logotip respecte a la versió anterior de l'aplicació el qual va quedar de la següent manera:



A3.1: Nou logotip de l'aplicació

A4. ALTRES PROTOTIPS

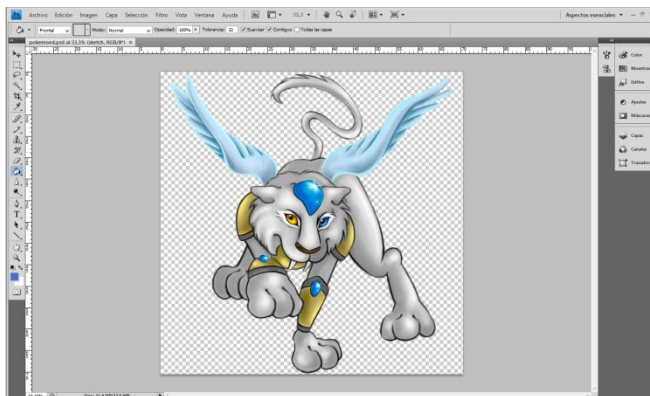
A continuació es pot veure el prototip dels monstres equipats:



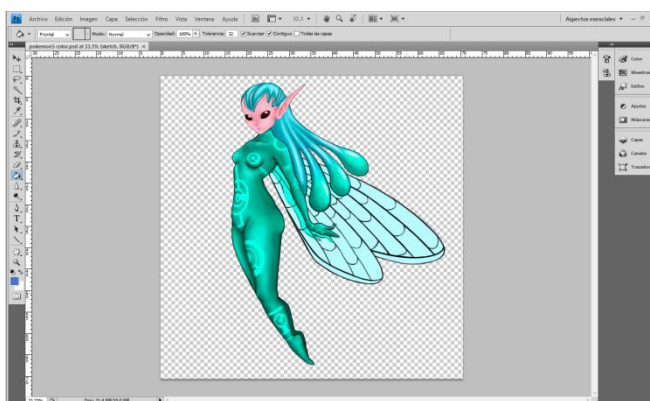
A4.1: Prototipus pantalla de monstres equipats

A5. ALTRES IL·LUSTRACIONS

A continuació es mostren altres il·lustracions que es van realitzar per a que el joc sigui el més original possible:



A5.1: Lleó blanc Tyr



A5.2: Fada Myrna



A5.2: Drag blau